



CAMPUS
DE EXCELENCIA
INTERNACIONAL



Graduado en Ingeniería Informática

Universidad Politécnica de Madrid
Facultad de Informática

TRABAJO FIN DE GRADO

Teller: Telemetría a partir de datos GPS

AUTOR: Sergio Gil Luque
TUTOR: Ángel Herranz Nieva
Madrid, Junio 2013

Índice general

Resumen	ix
Summary	xi
1 Introducción	1
1.1 Antecedentes	1
1.2 Motivación	2
1.3 Objetivos del proyecto	3
2 Análisis	5
2.1 Datos técnicos	5
2.1.1 Captura de datos	5
2.1.2 Programa de extracción de datos BT747	6
2.1.3 Formato CSV	6
2.2 Entendiendo los datos extraídos	8
2.2.1 Posición	8
2.2.2 Marca de tiempo	8
2.2.3 Distancia	8
2.2.4 Velocidad	8
2.2.5 Dirección	9
2.2.6 Otros campos	9

2.3	Entendiendo cada registro	10
2.4	Conceptos físicos	10
2.4.1	Operación con vectores	10
2.4.2	Conceptos telemétricos	13
2.4.3	Movimiento circular	15
2.5	Requisitos	17
3	Diseño	19
3.1	Obtención de nuevos datos	19
3.1.1	Obtención de la distancia	20
3.1.2	Obtención de la velocidad	20
3.1.3	Obtención de la dirección del recorrido	21
3.1.4	Obtención del radio de curvatura	21
3.1.5	Obtención de la aceleración	23
3.2	Clasificación de tramos	25
3.2.1	Clasificación usando el radio de curvatura	26
3.2.2	Clasificación mediante la variación en la dirección	26
3.3	Visualización	27
3.3.1	Fichero kml	27
3.3.2	Fichero html	29
3.4	Arquitectura del sistema	29
3.4.1	Diseño global	29
3.4.2	Módulos del programa	30
4	Ejecución de pruebas y resultados obtenidos	35
4.1	Ejecución de Teller	35
4.2	Información sobre las pruebas realizadas	37
4.3	Estudio sobre los resultados obtenidos	37

4.3.1	Distancia entre puntos	37
4.3.2	Velocidad	38
4.3.3	Dirección	39
4.3.4	Radio	40
4.4	Visualización de las pruebas	41
5	Conclusiones	49
5.1	Trabajo futuro	50
A	Dispositivo i-Blue 747A+	51
B	Listado de dispositivos GPS compatibles con BT747	55
	Bibliografía	57

Índice de figuras

2.1	Suma de vectores	12
2.2	Resta de vectores	12
2.3	Rotación de β grados	13
2.4	Vector posición	13
2.5	Vector desplazamiento	14
2.6	Movimiento circular uniforme	16
2.7	Movimiento circular no uniforme	17
3.1	Componentes de la aceleración en el punto P	23
3.2	Dos puntos P_1 y P_2 con sus respectivos vectores de velocidad	24
3.3	Dos puntos P_1 y P_2 con sus respectivos vectores de aceleración . . .	25
3.4	Ejemplo de capa KML en Google Maps	28
3.5	Módulos de Teller	30
3.6	Módulo csvParser	31
3.7	Módulo mathsGPS	31
3.8	Módulo classifier	32
3.9	Módulo kmlGenerator	33
4.1	Ejecución correcta de Teller	36
4.2	Trazado sin marcadores del recorrido 1	42
4.3	Trazado sin marcadores del recorrido 2	43

4.4	Trazado sin marcadores del recorrido 3	44
4.5	Trazado con marcadores del recorrido 1	45
4.6	Trazado con marcadores del recorrido 2	46
4.7	Trazado con marcadores del recorrido 3	47
A.1	Dispositivo i-Blue 747A+	51

Índice de cuadros

4.1	Tabla comparativa de distancias (Recorrido 2)	38
4.2	Tabla comparativa de velocidades (Recorrido 3)	39
4.3	Tabla comparativa de direcciones (Recorrido 3)	39
4.4	Radios de curvatura de puntos consecutivos (Recorrido 2)	41

Resumen

El avance y desarrollo de las nuevas tecnologías ha conseguido que los dispositivos GPS sean cada vez más efectivos y precisos. En la actualidad son usados por miles de personas en una gran variedad de aplicaciones, siendo la competición una de ellas. Detrás del funcionamiento de estas pequeñas unidades juega un papel muy importante la telemetría, una tecnología que cada vez es más relevante en el mundo actual, y que gracias a los avances tecnológicos sus aplicaciones son cada vez mayores.

En este Trabajo de Fin de Grado se diseñará *Teller*, un programa realizado en Erlang que mediante la telemetría y la tecnología actual realizará cálculos físicos de recorridos realizados en motocicleta, tales como velocidad o aceleración (tangencial y radial), además de clasificar los trayectos en tramos, dependiendo de si son curvas o rectas. Al mismo tiempo se realizará un estudio de precisión del dispositivo GPS, cotejando sus datos con los obtenidos por *Teller*.

Por último, toda esta información se mostrará en un mapa para que el usuario pueda visualizarlo y que de esta forma sirva como herramienta de información y aprendizaje.

Summary

The development of new technologies has contributed to the improvement of GPS devices, making them more effective and accurate. Nowadays they are used by thousands of people in a wide variety of applications like competitions. The technology that is behind the functionality of those devices is the telemetry, which is getting more and more important in the real world.

In this thesis a new program called *Teller* will be designed. Made in Erlang, it will use telemetry so as to obtain new and valuable data from a motorcycle track. Examples of this type of data can be speed and acceleration (either tangential and radial) for instance. Besides, *Teller* will classify the track into segments, depending if it is a curve or a straight. At the same time a study will be carried out in order to know how accurate the GPS device is. This will be achieved by comparing the data retrieved by the device with the data calculated by *Teller*.

Finally, all this information will be shown in a map, so the user can see information about his route in a visual way.

1

Introducción

El impacto de las nuevas tecnologías en el mundo actual es un hecho incuestionable que ha contribuido en gran medida a la mejora de la calidad de vida de la sociedad. La continua evolución de las comunicaciones inalámbricas, así como el abaratamiento de los dispositivos de posicionamiento global (del inglés *Global Positioning System* o *GPS*) son solo dos ejemplos que están muy relacionados con el campo de la telemetría. Gracias a su desarrollo constante, la importancia en el mundo actual de esta tecnología es cada vez mayor, así como las posibilidades que puede ofrecer de cara al futuro.

1.1 Antecedentes

La telemetría se puede definir como la tecnología que mide magnitudes físicas a distancia. Su origen se remonta a mediados del siglo XIX, aunque por aquel entonces la comunicación se realizaba mediante tecnología cableada. Como ejemplos se pueden destacar el desarrollo de un circuito que comunicaba el Palacio de Invierno del Zarato Ruso con el cuartel general del ejército en 1845, o la construcción en 1874 de un sistema meteorológico que calculaba el espesor de la nieve por medio de sensores. Sin embargo no sería hasta 1930, con la llegada de las primeras radiosondas, cuando la tecnología inalámbrica empezara a utilizarse en ese campo.

1 Introducción

En la actualidad, los usos de la telemetría son muy variados. Desde la supervisión y monitorización de grandes sistemas, pasando por su aplicación en el ámbito doméstico (como la medición de temperatura y energía) hasta su uso en competiciones profesionales. Como dos ejemplos importantes de este último caso se citan **MotoGP** y **Fórmula 1**, dos grandes eventos deportivos. Sin embargo, y continuando con este último caso, la telemetría también tiene su aplicación en un ámbito más generalista, en donde cualquier persona que disponga de una motocicleta y un dispositivo GPS pueda almacenar datos de los trayectos realizados.

La información que se almacena puede variar de un dispositivo a otro, pero hay datos que suelen ser comunes a todos, algunos porque son indispensables para este ámbito, y otros porque son muy útiles para el usuario. Entre otros podemos destacar la posición geográfica, la fecha en la que se ha tomado la muestra, velocidad, etc.

1.2 Motivación

Una de las características que tiene la telemetría en el contexto de las nuevas tecnologías es que aún tiene muchos campos por explorar e investigar. La utilidad básica de un dispositivo GPS (esto es, conocer la posición de un vehículo y el recorrido realizado) ya está suplida, al menos para la gran mayoría de usuarios que usan estos dispositivos diariamente. Sin embargo, la funcionalidad adicional que se le puede dotar al sistema es muy grande, más aún ahora que tienen una eficiencia mucho mayor a la de hace unos años.

Sin salir del mundo del motociclismo, hay un conjunto de factores físicos muy interesantes que intervienen en estos vehículos y que hacen que estos actúen de forma diferente. Sin ir más lejos, cuando una motocicleta toma una curva esta tiene que inclinarse debido a las fuerzas que actúan en ese momento. Actualmente todos estos datos, o al menos una gran mayoría de ellos pueden ser capturados mediante el uso de equipos profesionales que suelen usarse en las competiciones y que debido a su alto precio no son accesibles para los pilotos amateurs. Por lo tanto uno de los objetivos principales de este proyecto es poder obtener estos datos complejos teniendo como punto de partida un dispositivo GPS básico, y que mediante información telemétrica sencilla y usando conceptos físicos y matemáticos puedan conseguirse los mismos datos que actualmente solo se pueden obtener con equipos complejos.

Si bien es cierto que una mayor funcionalidad puede implicar mayor comple-

1.3 Objetivos del proyecto

jidad, no es en cualquier caso un motivo para no seguir investigando en este campo. Aparte de obtener más información para el usuario mediante la telemetría, los resultados y avances que se obtengan serán útiles y sentarán las bases para futuros estudios o mejoras que se quieran realizar en este campo.

Otro de los motivos que ha influenciado en la decisión de realizar este proyecto es la creciente popularidad que están teniendo los dispositivos GPS. En parte esto es gracias a la revolución que han causado los *smartphones*, teléfono móviles que ya tienen instalado un receptor GPS. Si bien es cierto que el presente proyecto excluye este tipo de dispositivos por cuestiones de tiempo y recursos, sin duda serán tenidos en cuenta en un futuro próximo, ya que si se instalan en un vehículo tendrían características similares a aquellos que están siendo considerados para este estudio.

Por último cabe destacar que con la realización de este proyecto los usuarios podrían comparar sus resultados con los de otros pilotos, y así servir como motivación para ellos, aparte de lograr un sistema más interactivo y atractivo.

1.3 Objetivos del proyecto

El principal objetivo de este proyecto es aprovechar las posibilidades que ofrece la tecnología actual en el contexto de la telemetría. Para eso se diseñará un programa que mediante los datos recogidos por un dispositivo GPS realice un conjunto de operaciones que sirvan para dar más información al usuario. Este programa, que se realizará usando el lenguaje de programación *Erlang*, recibirá el nombre de *Teller*.

Para este proyecto será indispensable el dispositivo GPS, encargado de recoger todos los datos del vehículo en el que esté instalado. Si bien como vehículo cualquiera sería apto para este proyecto se ha optado por considerar únicamente las **moto-cicletas**. Los motivos son varios: Por un lado no se podría profundizar mucho en el estudio si se tuvieran que considerar todos los tipos de vehículos en los que se puede usar un dispositivo GPS, ya que cada uno tiene sus características particulares que le diferencian del resto; por lo que los datos que se podrían extraer tendrían que ser aquellos que todos comparten, siendo los resultados muy básicos. Por otro lado, la elección de este tipo de vehículo no es aleatoria, sino que se eligió para poder realizar un interesante estudio sobre sus características físicas, así como su comportamiento. Es decir, aparte de calcular valores tales como distancia entre puntos o velocidad, también es posible realizar cálculos más complejos como aceleraciones que intervienen en las curvas, fuerzas, inclinación,

1 Introducción

radios de curvatura, etc...

Para poder estudiar estos comportamientos es indispensable realizar una clasificación de tramos. La más sencilla consiste en la diferenciación entre curvas y rectas, aunque dependiendo de la curvatura del tramo se podría realizar una clasificación más extensa dependiendo de los criterios que se sigan.

Todas las operaciones realizadas también pueden aprovecharse para ofrecer información visual. La visualización en un mapa del recorrido realizado, así como de los datos relevantes en cada punto (velocidad, aceleración, inclinación, etc), son opciones muy interesantes. No solo para el usuario de la motocicleta, que podrá comprobar los datos de su propio recorrido; sino para realizar estudios de comportamiento o de características similares.

Algunos de los datos que se calculen ya los habrá obtenido el propio dispositivo GPS también, aunque no se conocen los cálculos realizados por este. Por lo tanto se aprovechará esta situación para cotejar los datos que proporcione el dispositivo GPS con los de *Teller*, y de esta forma comprobar el grado de precisión y fiabilidad que este tiene. Para este cometido se ha elegido el modelo de GPS **i-Blue 747A+**. Esto es debido a que cada uno tiene sus propias características particulares, ya sea *hardware* (Reloj interno), como software (Algoritmos, lenguaje en el que se ha programado...), y por lo tanto los resultados obtenidos diferirían de forma significativa entre ellos.

2

Análisis

En este capítulo se documenta la fase de análisis del proyecto. Se trata de un estudio ligero pero bastante preciso, tanto de los datos proporcionados por el dispositivo GPS como de la información adicional que se derivará de estos.

2.1 Datos técnicos

Para poder realizar el proyecto es preciso conocer en detalle la información técnica del dispositivo GPS utilizado, así como las herramientas necesarias para descargar los datos y el formato utilizado. Todas estas características serán las que se traten a continuación.

Como se ha mencionado anteriormente, el punto de partida para nuestro programa es la información almacenada por un dispositivo registrador de datos (*data logger*) GPS. Se puede encontrar información sobre el dispositivo utilizado en este proyecto en el apéndice A.

2.1.1 Captura de datos

Por lo general, un dispositivo GPS realiza una captura de datos cada cierto tiempo. Este tiempo varía pero suele ser bastante pequeño, del orden de milisegun-

2 Análisis

dos (ms). En el caso particular de la unidad **i-Blue 747A+** la frecuencia de toma de datos es de 5 HZ ya que realiza capturas de datos 5 veces por segundo, es decir, cada 200 ms.

Es importante destacar que la frecuencia del dispositivo es un aspecto muy importante para el proyecto, ya que **se requiere que sea de alta frecuencia** para que los datos que se calculen sean lo más precisos posibles.

Los datos que un dispositivo GPS captura varían de un modelo a otro, pero algunos de esos datos son comunes a todos ellos. Se pueden mencionar: Latitud, longitud y marca de tiempo o *timestamp* entre otros.

En la sección 2.1.3 se definirán los campos que será necesario conocer para el correcto desarrollo del proyecto, y por tanto los campos que habrá que descargar del propio dispositivo para que *Teller* los procese.

2.1.2 Programa de extracción de datos BT747

Para poder realizar la extracción de los datos registrados por la unidad será necesario utilizar el programa *BT747* (<http://www.bt747.org/>). Este programa será el encargado de descargar los datos desde el dispositivo GPS al PC, pudiendo seleccionar distintos formatos de salida tales como *csv*, *gpx* y *NMEA* entre otros; siendo el primer formato el que se utilice como dato de entrada para este proyecto.

Este programa es compatible con la mayoría de dispositivos GPS que tienen incorporado un chip *MTK* o *MTK II*, por lo tanto abarca un amplio abanico de dispositivos que son compatibles con *Teller*. En el apéndice se muestra una lista de dispositivos compatibles con el programa.

El fichero que se obtiene después de que el programa *BT747* haya finalizado la descarga de datos del dispositivo GPS también se llama *log* o *track* y será el que reciba *Teller* como entrada. Sin embargo, por tratarse de la versión inicial del programa, **el log estará limitado al formato csv**, del que se pasará a hablar a continuación.

2.1.3 Formato CSV

Como ya se comentó anteriormente, el programa *BT747* ofrece la posibilidad de descargar los datos del GPS en diferentes formatos, siendo *csv* el que se usará para este proyecto. En cualquier caso cabe la posibilidad de que en un futuro se amplíe la compatibilidad a otros formatos ampliamente usados como son *gps* y

2.1 Datos técnicos

NMEA entre otros.

El nombre csv proviene del inglés *comma-separated values* (*Valores separados por comas*), y se trata de un formato sencillo en el que los valores se representan en forma de tabla, usando comas para diferenciar las columnas, y saltos de línea para cambiar de fila.

Como ya se comentó anteriormente, la información que se descarga del dispositivo es un *track* que contiene *trackpoints*. Los trackpoints son puntos o vectores de información que el dispositivo GPS captura cada cierto tiempo. La frecuencia de toma de datos, así como los campos que almacena depende de cada dispositivo y también de la configuración del programa *BT747*.

A continuación se muestran los campos que aparecen en el fichero *csv* para el caso particular de este proyecto, junto con una breve descripción de su significado. En el siguiente apartado se entrará más en detalle en cada uno de ellos.

- Index: Numeración del trackpoint
- RCR: Siglas en inglés de *ReCORD Reason*
- Date: Fecha en la que se tomó la muestra en formato *AAAA/MM/DD*
- Time: Hora en la que se tomó la muestra en formato *HH:MM:SS:ss*
- Valid: Indica el tipo de ajuste que se realizó para mejorar la precisión de la captura
- Latitude: Latitud (en grados)
- Longitude: Longitud (en grados)
- N/S: Indica si se trata del hemisferio norte (N) o sur (S)
- E/W: Indica si se trata del hemisferio este (E) u oeste (W)
- Height: Altura del punto con respecto al nivel del mar (en metros)
- Speed: Velocidad medida en kilómetros por hora (km/h)
- Heading: Dirección en la que se está viajando (en grados) .
- HDOP: Siglas en inglés de *Horizontal Dilution Of Precision*. Este concepto se explicará en el siguiente apartado
- Distance: Distancia entre el punto actual y el anterior (en metros)

2 Análisis

2.2 Entendiendo los datos extraídos

Una vez se conocen los campos del *track* se procederá a entrar en detalle en cada uno de ellos. Si bien algunos son triviales y no aportan mucha información, la mayoría serán muy útiles para entender el funcionamiento interno del dispositivo.

2.2.1 Posición

La posición del vehículo en coordenadas geográficas se obtiene con los campos *latitude*, *longitude*, *N/S* y *E/W*. Será uno de los principales datos en los que se basará *Teller* para realizar los posteriores cálculos.

2.2.2 Marca de tiempo

La marca de tiempo o *timestamp* se obtiene con los campos *Date* y *Time*. De esta forma se puede saber cuanto tiempo ha transcurrido desde un punto P_i a otro P_{i+1} y así detectar posibles lapsos grandes de tiempo en los que no se haya podido realizar una captura de datos por diversos motivos, como por ejemplo problemas de conexión con los satélites.

2.2.3 Distancia

El campo *Distance* indica los metros recorridos desde el punto anterior al actual. Para comprobar la precisión del dispositivo, *Teller* calculará también la distancia con el punto anterior, basándose únicamente en las coordenadas geográficas proporcionadas por el GPS. La forma de calcularlo se explica en el capítulo 3.1

2.2.4 Velocidad

El propio dispositivo GPS también proporciona la velocidad de la motocicleta en un punto mediante el campo *Speed*, medido en km/h. Al igual que con la distancia, *Teller* recalculará este dato usando las coordenadas geográficas proporcionadas por el GPS.

2.2 Entendiendo los datos extraídos

2.2.5 Dirección

La dirección del vehículo viene definida por el campo *Heading*. Está medido en grados siendo 0 dirección norte y continuando en sentido horario. Por lo tanto 90 grados corresponderán a dirección este, 180 a dirección sur, y 270 a dirección oeste.

Aunque este valor no se determina magnéticamente, se utilizan diversas técnicas como el *efecto doppler*. Al igual que en los apartados anteriores, *Teller* calculará este dato de forma independiente.

2.2.6 Otros campos

A continuación se detallan en menor medida los otros datos que proporciona el dispositivo GPS pero que no serán muy útiles en cuanto a objetivos del proyecto se refiere:

- **Altura:** El campo *Height* indica la altura en metros del vehículo respecto al nivel del mar. Debido a que esta información no se va a tener en cuenta a la hora de realizar los cálculos (es decir, se considera que no hay diferencia de altura en todo el trayecto), este campo no se utilizará y puede ser descartado.
- **Index:** El índice del punto tampoco será tan importante como los otros datos mencionados. La numeración de los puntos será secuencial incluso en los momentos en los que no se ha podido realizar la toma de datos con la frecuencia indicada (Por ejemplo, si un dispositivo GPS con una frecuencia de captura de 200 ms hay una diferencia de 400 ms o más en alguna de ellas). Por lo tanto este campo solo será útil para numerar los puntos a la hora de visualizar los datos así como para tareas de depuración.
- **HDOP, Valid, RCR:** Estos campos indican en mayor o menor medida el tipo de precisión que tiene el dispositivo GPS en un momento concreto. Si por ejemplo el vehículo realiza un recorrido en un día muy nublado, o por cualquier circunstancia no puede establecer conexión con suficientes satélites, la precisión se verá afectada.

Estos datos actualmente son irrelevantes para el proyecto y no serán utilizados por *Teller*.

2 Análisis

2.3 Entendiendo cada registro

Los campos detallados en la sección anterior son los que conformarán cada captura que realice el dispositivo GPS. Como ya se explicó anteriormente, estos datos, llamados *trackpoints*, estarán almacenados en formato *csv*.

El motivo por el que se usa el término *trackpoint* en vez de punto GPS o registro es debido a que la cantidad de información que se obtiene por cada captura no se ciñe solo a la posición geográfica, sino a datos como la distancia, velocidad o dirección. No sería por lo tanto correcto decir que se está almacenando un punto de información GPS, ni siquiera un registro. El término *trackpoint* o vector de información es el más acertado para este caso, y el que se ha usado en el proyecto.

2.4 Conceptos físicos

En este apartado se tratarán aquellos conceptos físicos que serán útiles para el desarrollo del presente proyecto. En términos generales se puede definir una motocicleta como una partícula en movimiento que, como toda partícula, está sujeta a distintas fuerzas y magnitudes que se pueden representar mediante vectores.

Un vector es una herramienta geométrica que representa magnitudes físicas por medio de su módulo, dirección y sentido. En física son muchas las magnitudes que se pueden representar mediante estos parámetros. Entre otras se encuentran el desplazamiento, la velocidad y la aceleración de una partícula, así como las fuerzas que intervienen sobre ella. Es esencial para este proyecto conocer y representar estas magnitudes mediante vectores, por lo que a continuación se pasará a entrar en detalle en estos conceptos.

2.4.1 Operación con vectores

A continuación se repasan los conceptos básicos de operaciones con vectores que serán muy útiles para el desarrollo del proyecto.

Coordenadas cartesianas y polares

Dependiendo del sistema de coordenadas en el que se esté trabajando, algunas operaciones con vectores serán más sencillas de realizar que otras. Para este

2.4 Conceptos físicos

proyecto se trabajará con dos: Coordenadas cartesianas y coordenadas polares.

Como breve recordatorio cabe mencionar que las coordenadas cartesianas son de la forma $A = (xi, yj)$, mientras que las polares son de la forma $B = (r, \theta)$.

Conversión de coordenadas

- De coordenadas cartesianas a polares

Teniendo en coordenadas cartesianas el punto $A = (x, y)$, el cálculo de r y θ se obtiene de la siguiente manera:

$$r = \sqrt{x^2 + y^2}$$

$$\theta = \arctan\left(\frac{y}{x}\right)$$

- De coordenadas polares a cartesianas

Teniendo en coordenadas polares el punto $A' = (r, \theta)$, la conversión a coordenadas cartesianas se realiza como se muestra a continuación:

$$x = r \cos \theta$$

$$y = r \sin \theta$$

Suma y resta de vectores

La suma de un vector \vec{a} y otro \vec{b} da como resultado un tercer vector \vec{c} . En coordenadas cartesianas resulta de la suma de los coordenadas de los vectores.

$$\vec{a} = (x_1, y_1)$$

$$\vec{b} = (x_2, y_2)$$

$$\vec{c} = \vec{a} + \vec{b} = (x_1 + x_2, y_1 + y_2)$$

En el caso de la resta se trata de la suma de \vec{a} con el opuesto de \vec{b} al que llamaremos $\vec{\bar{b}}$

2 Análisis

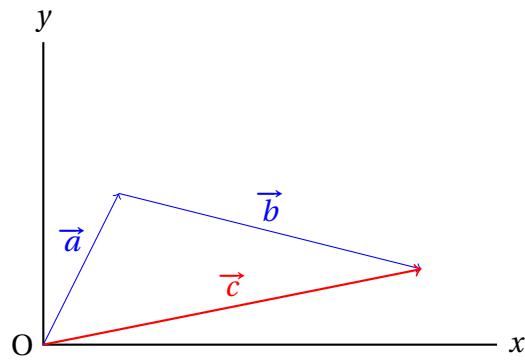


Figura 2.1: Suma de vectores

$$\vec{a} = (x_1, y_1)$$

$$\vec{b} = (x_2, y_2)$$

$$\vec{b}' = (-x_2, -y_2)$$

$$\vec{c} = \vec{a} - \vec{b} = \vec{a} + \vec{b}'$$

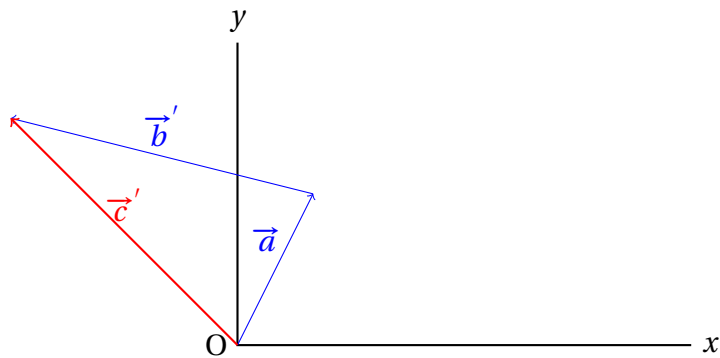


Figura 2.2: Resta de vectores

Rotación de vectores

Rotar vectores en coordenadas polares es una operación sencilla. Basta con sumar o restar los grados que se quieren rotar del vector en cuestión.

2.4 Conceptos físicos

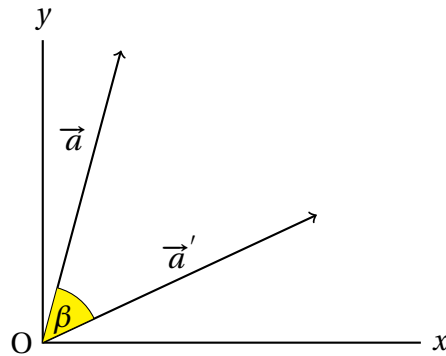


Figura 2.3: Rotación de β grados

2.4.2 Conceptos telemétricos

Para este proyecto es indispensable tener información básica sobre el vehículo que permita conocer su posición, velocidad a la que se desplaza, aceleración, etc. Los siguientes conceptos que se detallan a continuación son básicos para poder trabajar en este campo.

Vector posición: Consiste en un vector trazado desde el origen de coordenadas hasta un punto P . Así por ejemplo, una partícula que se encuentre en un punto con coordenadas (x, y) tendrá como vector posición el que se muestra a continuación

$$\vec{r} = xi + yj$$

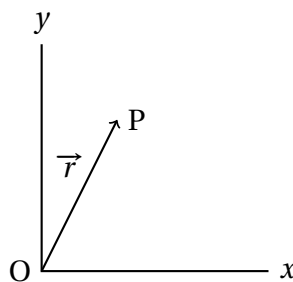


Figura 2.4: Vector posición

Vector desplazamiento: Es un vector que define el desplazamiento de una partícula desde un punto inicial P_1 a otro punto P_2 .

2 Análisis

$$\Delta \vec{r} = \vec{r}_2 - \vec{r}_1$$

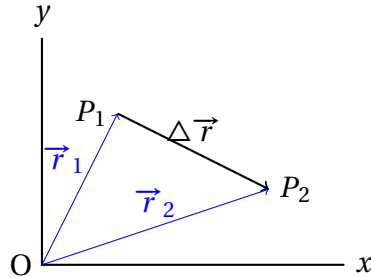


Figura 2.5: Vector desplazamiento

Una vez definidos los vectores de posición y desplazamiento se puede empezar a hablar del concepto de velocidad, definido como la cantidad de espacio recorrido por la partícula por unidad de tiempo. Las dos magnitudes vectoriales que más interesan para el proyecto en cuestión son: **Velocidad media** y **velocidad instantánea**.

Velocidad media: La velocidad media de una partícula se obtiene dividiendo el vector desplazamiento entre dos puntos entre el intervalo de tiempo transcurrido. Este vector tiene la misma dirección y sentido que el de desplazamiento.

$$\vec{v}_m = \frac{\Delta \vec{r}}{\Delta t}$$

Velocidad instantánea: Si se quiere conocer la velocidad de una partícula en un instante concreto de su trayectoria se calcula la velocidad instantánea. A diferencia de la velocidad media, este vector se calcula cuando Δt tiende a 0, o lo que es lo mismo

$$\vec{v}_{inst} = \lim_{\Delta t \rightarrow 0} \frac{\Delta \vec{r}}{\Delta t}$$

La dirección y sentido de este vector es la de la tangente de la trayectoria de la partícula.

Otra magnitud vectorial que es esencial para este proyecto es la **aceleración**. Se define como el cambio de la velocidad por unidad de tiempo. Y al igual que en esta, en la aceleración existen varias magnitudes vectoriales que serán muy utilizadas de ahora en adelante.

2.4 Conceptos físicos

Aceleración media: Como ya se ha mencionado anteriormente, una partícula tendrá una componente de velocidad instantánea \vec{v}_{inst} en cada instante de tiempo de una trayectoria. Se define aceleración media como la variación de la velocidad instantánea en el tiempo.

$$\vec{a}_{media} = \frac{\Delta \vec{v}}{\Delta t}$$

Aceleración instantánea: Siguiendo el mismo procedimiento para el cálculo de la velocidad instantánea, la aceleración instantánea se define como el límite de la aceleración media cuando el intervalo de tiempo tiende a 0.

$$\vec{a}_{inst} = \lim_{\Delta t \rightarrow 0} \frac{\Delta \vec{v}}{\Delta t}$$

2.4.3 Movimiento circular

Cuando una partícula se está moviendo a lo largo de una trayectoria circular la dirección de su velocidad varía, lo que quiere decir que hay una componente de la aceleración perpendicular a su trayectoria. Se trata de un **movimiento circular uniforme** ya que el tiempo que tarda en recorrer una determinada distancia es el mismo en cualquier punto del círculo.

Al variar la dirección de la velocidad tiene que haber una componente de la aceleración que sea la que la modifique. Esta componente se denomina **aceleración centrípeta**, y apunta siempre al centro de la circunferencia, por lo que es perpendicular a la de la velocidad

El cálculo de la aceleración centrípeta se realiza de la siguiente manera:

$$\vec{a}_{cent} = \frac{v^2}{R}$$

Sin embargo este no es el tipo de movimiento que ocurre normalmente en la realidad. Volviendo al ámbito de la telemetría, una motocicleta varía su velocidad

2 Análisis

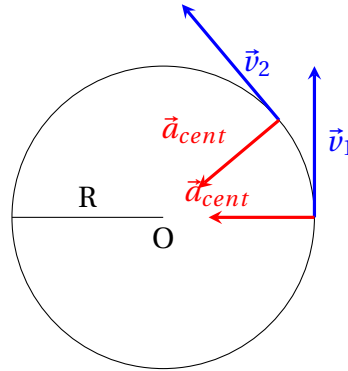


Figura 2.6: Movimiento circular uniforme

constantemente, especialmente en las curvas. En ese caso se está hablando del **movimiento circular no uniforme**, que será el que sirva para este proyecto.

Cuando una partícula se está desplazando por una curva con velocidad variable, existen dos componentes de la aceleración involucradas y perpendiculares entre sí: **La aceleración tangencial y la aceleración radial**.

La aceleración radial es igual que la aceleración centrípeta en el movimiento circular uniforme. Es la componente que apunta al centro de la circunferencia y la que provoca el cambio en la dirección de la velocidad. La ecuación que calcula su valor, al igual que en el caso anterior, es

$$\vec{a}_{rad} = \frac{v^2}{R}$$

Sin embargo en este tipo de movimiento la aceleración radial no es la única que actúa sobre la partícula. Hay otra componente que tiene la misma dirección que la de la velocidad de la partícula en el punto, y que provoca la variación del módulo de la velocidad durante su trayectoria. Esta componente de la aceleración se llama aceleración tangencial, por tener la misma dirección que la de la tangente a la curva.

Su valor es igual a la tasa de cambio de la velocidad, por lo tanto

$$\vec{a}_{tan} = \frac{d\vec{v}}{dt}$$

Al haber dos componentes de la aceleración que actúan sobre la partícula, la aceleración total será la suma de esos dos vectores. Usando el Teorema de Pitá-

2.5 Requisitos

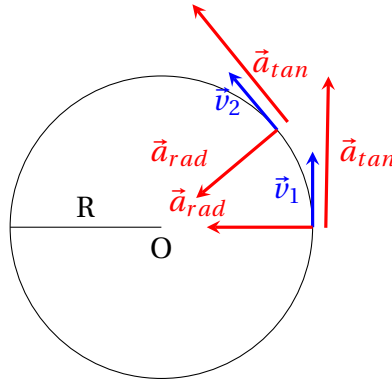


Figura 2.7: Movimiento circular no uniforme

goras tenemos

$$\vec{a}_{total} = \sqrt{\vec{a}_{tan}^2 + \vec{a}_{rad}^2}$$

A partir de estos datos puede obtenerse otro tipo de información de forma sencilla como pueden ser fuerzas que intervienen en las curvas, y a partir de estas calcular la inclinación que debería tener la moto en esta. Sin embargo esto no formará parte del proyecto ya que se quiere incidir más en otros aspectos como la clasificación de tramos o su visualización, y debido a la limitación de tiempo no es posible abarcar en profundidad todos ellos.

2.5 Requisitos

Con el análisis tanto de las características técnicas del dispositivo como de los conceptos físicos involucrados en este tipo de vehículos, se puede comenzar a definir las características del programa que se quiere crear para este proyecto. Como ya se ha mencionado anteriormente, el nombre de este programa será *Teller*, y tendrá que cumplir una serie de requisitos que se definen a continuación.

- Cálculo de información telemétrica

Cálculo de nuevos datos a partir de los proporcionados por el dispositivo GPS. Algunos de ellos ya los calcula el propio dispositivo, pero *Teller* los obtendrá de forma independiente para poder cotejarlos.

Los datos en cuestión son:

2 Análisis

- Distancia entre puntos
 - Dirección
 - Velocidad
 - Aceleración tangencial
 - Aceleración radial o centrípeta
- Clasificación del recorrido

Se realizará una clasificación básica del recorrido en curvas o rectas, teniendo en cuenta los datos capturados, así como la información adicional capturada por *Teller*
- Visualización

Por último todos los resultados obtenidos se mostrarán en un mapa para que el usuario pueda visualizarlos. El servicio que se utilizará para poder realizar esta tarea será Google Maps (<https://maps.google.com/>).

3

Diseño

Una vez se ha realizado la fase de análisis del proyecto se puede comenzar con el diseño del programa. Esto incluye el cálculo de nuevos parámetros, clasificación de los tramos y su posterior visualización en un mapa.

3.1 Obtención de nuevos datos

Una vez se ha realizado el estudio sobre los datos que proporciona el dispositivo GPS se puede comenzar con el cálculo y obtención de información adicional que se deriven de ellos. Parte de los datos que se van a calcular ya estarán almacenados por el propio dispositivo GPS, como por ejemplo la velocidad o la distancia entre puntos. Sin embargo, al no conocerse el método que el dispositivo usa internamente para obtener esa información tampoco se puede asegurar el grado de precisión que este proporciona. Por lo tanto, el objetivo de este cálculo es doble: Por un lado determinar de forma independiente datos telemétricos utilizando únicamente las coordenadas geográficas proporcionadas por el dispositivo GPS y por otro lado determinar, cotejando estos datos, el grado de precisión de este.

Cabe mencionar nuevamente que los datos más importantes que permitan realizar todos estos cálculos serán las coordenadas geográficas proporcionadas por la propia unidad. Puede resultar contraproducente realizar un estudio de

3 Diseño

precisión sobre un dispositivo GPS del cual se están usando parte de sus datos como base. Sin embargo, al tratarse de un dispositivo de posicionamiento se da por hecho que las coordenadas geográficas serán las más precisas por la importancia que tienen en este ámbito.

3.1.1 Obtención de la distancia

Como ya se vio en el capítulo 2.2, el propio dispositivo GPS, además de registrar información imprescindible como las coordenadas geográficas o la dirección del vehículo, también almacena la distancia en metros entre un punto y su anterior.

Para volver a calcular este valor conociendo solo las coordenadas geográficas de dos puntos se utilizará la fórmula **Haversine**. Esta fórmula calcula la distancia entre dos puntos de una esfera conociendo su longitud y latitud. La forma de calcularlo, teniendo un punto P_1 con longitud λ_1 y latitud ϕ_1 , y otro punto P_2 con longitud λ_2 y latitud ϕ_2 es la siguiente:

$$d = 2r \arcsin\left(\sqrt{\sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos(\phi_1) \cos(\phi_2) \sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right)$$

siendo d la distancia entre los dos puntos y r el radio de la Tierra.

Para poder cotejar este dato con el proporcionado por el propio dispositivo GPS se usará la misma unidad, es decir, el metro. Por lo tanto, el radio de la Tierra tendrá que proporcionarse en metros también. Obviamente este último dato variará según la altura en la que se encuentre el vehículo, ya que el radio no tendrá el mismo valor para un recorrido por la costa que uno por la montaña. En cualquier caso hay que recordar que la diferencia de altura entre puntos consecutivos será muy pequeña debido a la frecuencia de toma de datos, que será en la mayoría de los casos de 200 ms. Sin embargo, y como ya se comentó anteriormente, la altura no va a ser considerada para este proyecto.

3.1.2 Obtención de la velocidad

Una vez se ha obtenido la distancia entre dos puntos usando el método que se ha explicado en la sección anterior, y teniendo en cuenta la frecuencia con la que el dispositivo realiza una nueva captura de datos, el cálculo de la velocidad es sencillo:

$$v = \frac{d}{\Delta t}$$

3.1 Obtención de nuevos datos

donde d es la distancia entre los dos puntos, y Δt es el tiempo transcurrido desde el primer punto al segundo.

Una duda que se plantea en este punto es determinar si lo que se está calculando es la velocidad media o la velocidad instantánea. En el capítulo 2.4.2 se vieron las dos definiciones, y si hubiera que ceñirse estrictamente a la teoría entonces el cálculo que se estaría haciendo sería el de la velocidad media entre dos puntos P_1 y P_2 con un intervalo de tiempo Δt .

Sin embargo en la práctica sería más adecuado decir que se trata de velocidad instantánea ya que Δt es en la mayoría de los casos 200ms, un valor suficientemente pequeño como para considerar que se está calculando la velocidad instantánea en P_2 . Con esto se puede conseguir calcular la aceleración del vehículo ya que, como se puede comprobar nuevamente en el capítulo 2.4.2, se necesita la velocidad instantánea, y aunque no se pueda conseguir un valor exacto debido a que Δt no tiende a 0, se puede realizar una aproximación bastante aceptable del valor real.

3.1.3 Obtención de la dirección del recorrido

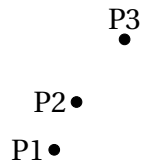
La dirección que tiene el vehículo en un punto determinado, también llamada *heading*, se puede calcular si se conocen las coordenadas geográficas de esos puntos. Al ser puntos muy cercanos entre si, el vector desplazamiento sirve también para obtener la dirección. También es cierto que habrá un pequeño margen de error al calcular estos datos ya que seguirá habiendo distancia entre cada par de puntos. En el capítulo 4 se cotejan estos datos con los del dispositivo GPS.

3.1.4 Obtención del radio de curvatura

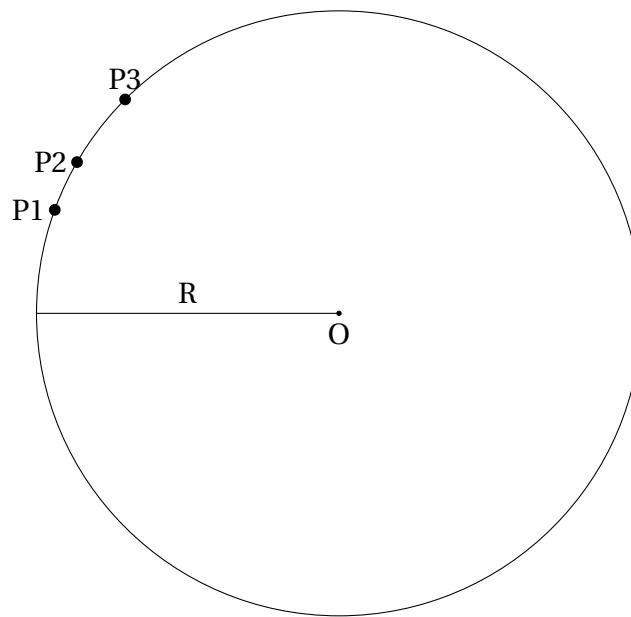
La obtención del radio de la curva que pasa por tres puntos es un dato muy interesante que puede servir para realizar otros cálculos más avanzados. **Sin embargo no se asegura que sea fiable**, ya que en trayectos reales las curvas que se realizan con un vehículo no tienen un radio constante, si no que varían constantemente. Habrá que observar por lo tanto los resultados que se obtengan para determinar si pueden ser útiles de cara a futuros cálculos.

El método que se ha seguido para calcular el radio es el siguiente: Sean P_1 , P_2 y P_3 tres puntos consecutivos del trayecto realizado por el vehículo, como se muestra a continuación:

3 Diseño



El radio de la curva que pasa por los tres puntos es precisamente el radio de la circunferencia circunscrita a ellos:



El calculo del radio de una circunferencia circunscrita es

$$R = \frac{d_1 d_2 d_3}{4K}$$

siendo K el área del triángulo formado por los tres puntos, y $d_1 d_2 d_3$ los lados de este. Estos lados ya se conocen porque d_1 corresponde al vector desplazamiento de P_1 a P_2 y d_2 al mismo tipo de vector de P_2 a P_3 . De igual forma, con esos datos se puede obtener d_3 y por lo tanto obtener el área por medio de la fórmula de Herón:

$$K = \sqrt{s(s - d_1)(s - d_2)(s - d_3)}$$

siendo s el semiperímetro del triángulo, dato que también se puede obtener:

$$s = \frac{d_1 + d_2 + d_3}{2}$$

3.1.5 Obtención de la aceleración

Al igual que se ha hecho en el anterior apartado, se considerará que la aceleración que se va a calcular a continuación será la aceleración instantánea en un punto por ser el intervalo de tiempo lo suficientemente pequeño; aunque teóricamente se está calculando la aceleración media entre dos puntos.

Los datos que hacen falta para calcular la aceleración entre dos puntos P_1 y P_2 ya se conocen o han sido calculados previamente. Estos datos son la velocidad instantánea en ambos puntos, y el intervalo de tiempo Δt que se ha tardado en ir del punto P_1 al punto P_2 . La fórmula sería:

$$a = \frac{v_2 - v_1}{\Delta t}$$

Obtención de la aceleración radial y tangencial

Retomando lo que se explicó en el capítulo 2.4, la aceleración solo tiene una única componente en los movimientos rectilíneos, y dos en los circulares. En este último caso existe la aceleración tangencial, con dirección tangente a la circunferencia en el punto, y la aceleración radial, con dirección el centro de la circunferencia que simula la curva.

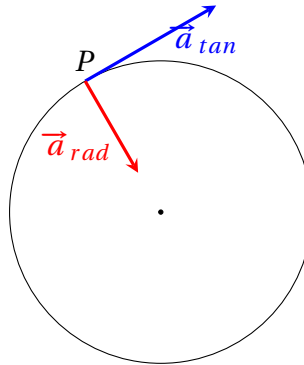


Figura 3.1: Componentes de la aceleración en el punto P

Los datos de los que se dispone son dos: Por un lado la diferencia de velocidades en un punto Δv y por otro lado el radio de curvatura de tres puntos (el punto actual, el anterior y el siguiente). Con estos datos hay dos formas de calcular las componentes de la aceleración. Una de ellas necesita el radio de curvatura en el punto, y la otra hace uso de la trigonometría. A continuación se explicarán ambos métodos:

3 Diseño

- Usando el radio: En el capítulo 2.4 se mostraron las fórmulas para calcular la aceleración radial y la tangencial. Estas son:

$$\vec{a}_{rad} = \frac{v^2}{R}$$

$$\vec{a}_{tan} = \frac{d\vec{v}}{dt}$$

Como se puede observar, el cálculo de la aceleración tangencial se obtiene fácilmente si se conoce la velocidad instantánea. Lo mismo ocurre para el caso de la aceleración radial si se conoce el radio de curvatura.

- Sin usar el radio: Si no se conoce el radio de curvatura la aceleración radial no puede obtenerse mediante la fórmula anterior. Sin embargo hay otro método más largo con el que se puede calcular si se conoce la velocidad del vehículo en los dos puntos.

Sean dos puntos P_1 y P_2 , y sus vectores de velocidad \vec{v}_1 y \vec{v}_2 mostrados en la figura 3.2.

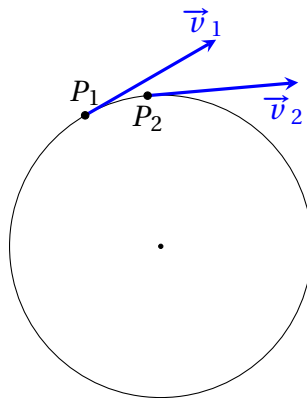


Figura 3.2: Dos puntos P_1 y P_2 con sus respectivos vectores de velocidad

Los vectores tendrán como dirección la de la trayectoria del vehículo en ese punto. Por tratarse de un movimiento circular hay dos componentes de la aceleración que intervienen: Por un lado la aceleración radial, que modifica la dirección del vector de velocidad, y por otro lado la aceleración tangencial, que modifica el módulo de la velocidad (y que sería 0 si no hubiera ninguna modificación).

3.2 Clasificación de tramos

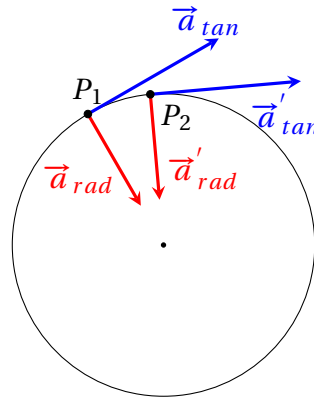
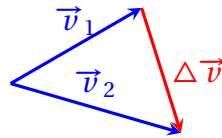


Figura 3.3: Dos puntos P_1 y P_2 con sus respectivos vectores de aceleración

Si nos centramos exclusivamente en los vectores de velocidad se puede comprobar que existe una variación de velocidad $\Delta \vec{v}$



Es sencillo obtener las componentes x e y de $\Delta \vec{v}$, que son al mismo tiempo la velocidad tangencial y la velocidad radial respectivamente.

Por último, para obtener la aceleración radial y tangencial se divide cada respectiva componente de la velocidad obtenida por el tiempo transcurrido entre los dos puntos.

3.2 Clasificación de tramos

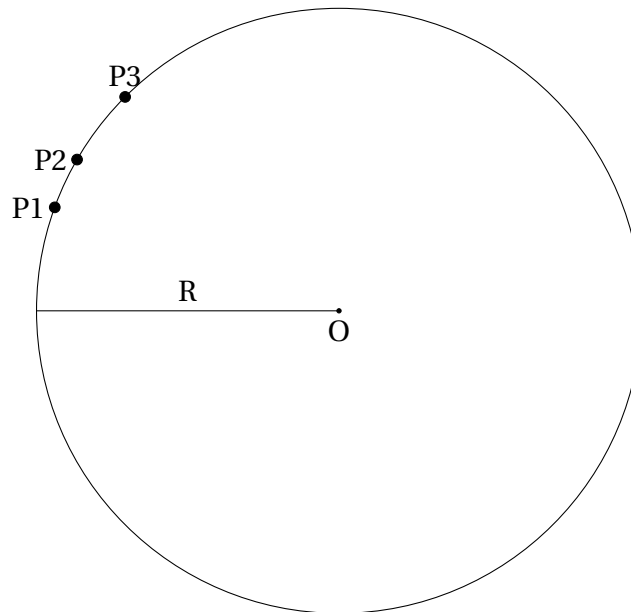
Otra de las funcionalidades que tendrá *Teller* será la clasificación de trayectos en tramos o segmentos. La clasificación básica consistirá en curvas o rectas, si bien estos conceptos habrá que estudiarlos más en profundidad para definir exactamente qué es una curva y qué es una recta.

Teniendo en cuenta que los datos con los que se están trabajando hay varias formas de abordar este problema

3 Diseño

3.2.1 Clasificación usando el radio de curvatura

Uno de los datos que *Teller* calcula y almacena es el radio de la curva que pasa por tres puntos consecutivos. Considérese tres puntos P_1, P_2, P_3 consecutivos correspondientes a tres tomas de datos del dispositivo GPS, y R el radio de la circunferencia circunscrita a ellos



Como puede comprobarse, cuanto más cerrada es la curva que la moto está tomando, menor es el radio de la circunferencia circunscrita. Por otro lado, cuanto menos curva sea, mayor será el radio. El caso más extremo sería una línea recta formada por los tres puntos, lo que daría un radio infinito.

Por lo tanto, si se computan los radios de todos los puntos, se podría hacer una clasificación de tramos, dependiendo del valor de R obtenido.

Sin embargo, y como ya se mencionó en el capítulo 3.1.4, este dato no se puede confirmar con seguridad que sea fiable hasta que se realicen las pruebas y se comprueben los resultados obtenidos.

3.2.2 Clasificación mediante la variación en la dirección

Si no se dispone del radio de curvatura o este no es un dato lo suficientemente fiable, hay otra forma con la que realizar la clasificación. Como cada punto tiene

3.3 Visualización

información sobre la dirección o *heading* que está tomando el vehículo, la variación de este dato entre un punto y el siguiente puede usarse para conocer el grado de curvatura.

Por lo tanto podría realizarse una clasificación entre curvas y rectas si la variación en la dirección entre un punto y el siguiente supera o no un determinado valor. Para las pruebas que se van a realizar con *Teller* en este proyecto este valor será de 1 grado. Es decir, si la variación en la dirección entre dos puntos es mayor a 1 grado se considerará una curva, y en caso contrario una recta.

3.3 Visualización

Una vez se han obtenido todos los datos necesarios, así como la clasificación de los tramos del trayecto proporcionado, es necesario mostrárselos al usuario para que pueda visualizarlos.

Como ya se mencionó anteriormente, para realizar la visualización se utilizará el servicio web de *Google Maps*, con el que se pueden mostrar información sobre una posición dada, así como trazar recorridos mediante el uso de capas o *layers*, concepto que se explicará en esta sección.

Para conocer más a fondo el funcionamiento de este servicio así como la forma de poder visualizar estos datos, se ha estudiado la API de Google Maps versión 3 (<https://developers.google.com/maps/>).

En términos generales, para visualizar los datos calculados por *Teller* hacen falta generar dos ficheros: Un fichero **kml** y otro **html**. A continuación se define la función que tiene cada uno.

3.3.1 Fichero kml

Un fichero *kml* (acrónimo en inglés de *Keyhole Markup Language*) es un formato basado en *XML* para representar datos geográficos. Están diseñados especialmente para *Google Earth* y *Google Maps*, los cuales son muy utilizados. *Teller* generará este fichero al finalizar la ejecución, en donde estará almacenada toda la información que más adelante se visualizará. Es importante destacar que el fichero *kml* tiene que tener una dirección pública para que *Google Maps* pueda acceder sin problemas. En caso contrario no se podrá realizar la visualización.

El contenido de este fichero se puede clasificar en dos grandes grupos: **La capa KML y los marcadores**.

3 Diseño

Capa KML

Más conocida por *KML Layer*, se trata de una capa que se superpone sobre el mapa que se va a visualizar, y que contiene información personalizada. Por lo tanto se puede utilizar esta capa para visualizar la clasificación de tramos realizado por *Teller*. Los criterios son los siguientes:

- **Trazado azul:** Tramos de recta (Variación en la dirección menor o igual a un grado)
- **Trazado rojo:** Tramos de curva (Variación en la dirección mayor a un grado)

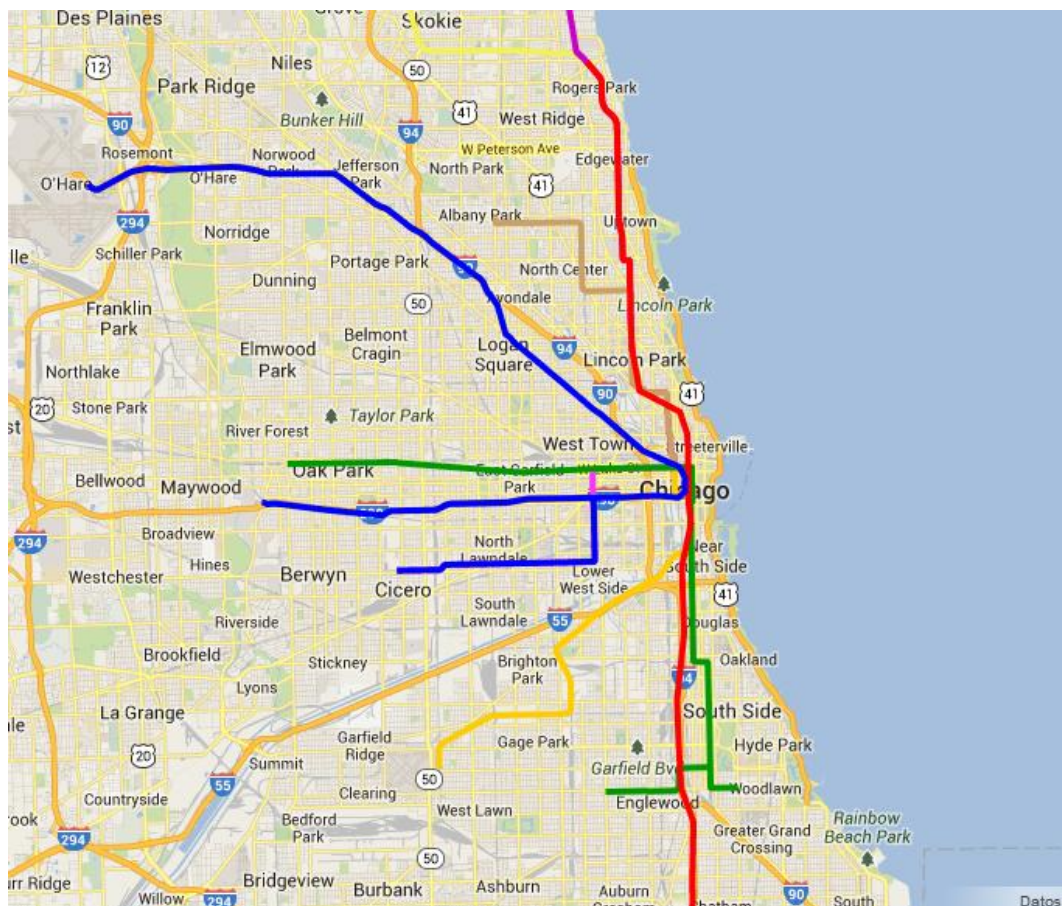


Figura 3.4: Ejemplo de capa KML en Google Maps

3.4 Arquitectura del sistema

Marcadores

Los marcadores o *Markers* son puntos en un mapa que contienen información. Para este proyecto es un elemento muy útil ya que es posible colocar un marcador por cada captura de datos realizada por el dispositivo GPS, mostrando información calculada por *Teller*. Para esta versión del programa los datos que se mostrarán serán los siguientes:

- Numeración interna del punto (Index)
- Velocidad
- Aceleración total
- Aceleración tangencial
- Aceleración radial

3.3.2 Fichero html

Este fichero será el que genere *Teller* al finalizar la ejecución del programa y que se conectará con *Google Maps*, indicándole la dirección del fichero *kml* para poder visualizar los resultados obtenidos.

3.4 Arquitectura del sistema

A continuación se presenta la arquitectura de *Teller* una vez conocidos a fondo los requisitos necesarios, tanto de obtención de datos como de clasificación y visualización de estos.

3.4.1 Diseño global

El diseño de la arquitectura del sistema ha sufrido importantes cambios desde su concepción inicial hasta su versión final. Sin embargo esto era algo que ya estaba planeado en cierto modo, ya que se ha adoptado la metodología de desarrollo de software *Extreme Programming* (o simplemente *XP*). Esta metodología, que forma parte de los procesos ágiles de desarrollo de software, consiste en ir modificando los requisitos mientras el proyecto va avanzando si la situación así

3 Diseño

lo requiere. Con esto se consigue agilizar la programación sin perder demasiado tiempo en definir todos los requisitos del proyecto.

La arquitectura final de Teller es la que se muestra en la figura 3.5. En la siguiente sección se pasará a entrar en detalle en cada uno de los módulos que componen el programa.

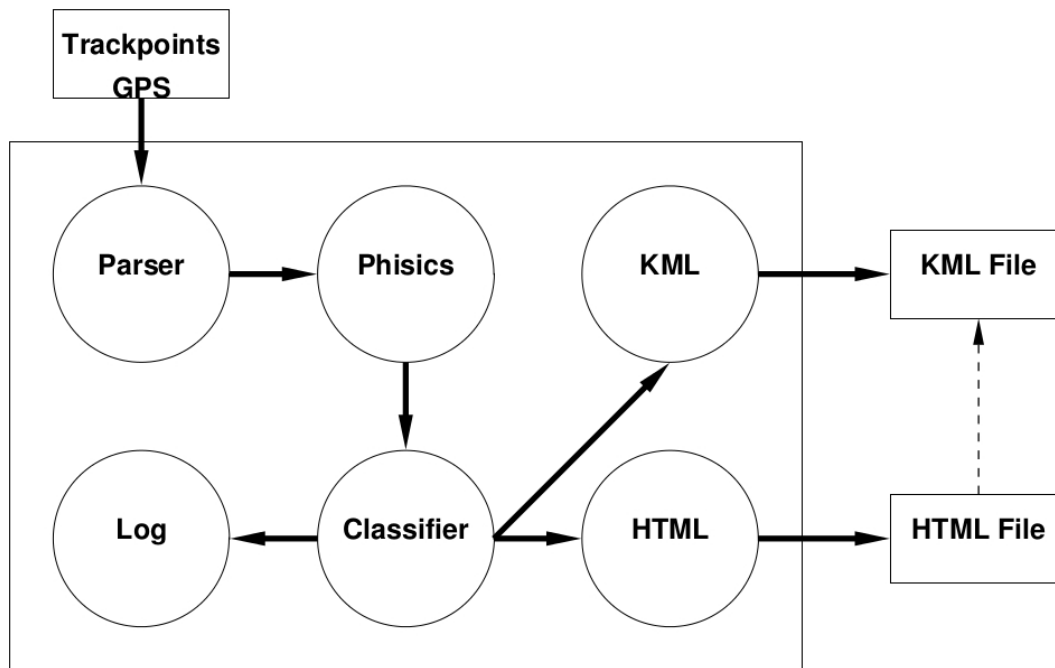


Figura 3.5: Módulos de Teller

3.4.2 Módulos del programa

Teller, en su diseño final, consta de 8 módulos, todos programados en Erlang y cada uno con una funcionalidad específica. A continuación se procederá a enumerar cada uno de ellos, definiendo sus características.

Módulo initializer

Es el módulo principal, que recibirá como entrada los datos capturados por el dispositivo GPS en formato csv y se irá comunicando con el resto de módulos del programa para obtener todos los datos adicionales necesarios, así como los ficheros de clasificación y visualización del recorrido.

3.4 Arquitectura del sistema

Módulo csvParser

El módulo *csvParser* será el que reciba el fichero csv directamente del módulo *initializer* al inicio del programa. Su función consiste en transformar esos datos, que inicialmente serán valores separados por comas, en una lista de *trackpoints*. Al estar programado en un lenguaje funcional como *Erlang*, trabajar con listas es mucho más rápido y cómodo.



Figura 3.6: Módulo csvParser

Módulo mathsGPS

Es el módulo que realiza todos los cálculos matemáticos para obtener nuevos datos. Entre ellos se pueden destacar distancia entre puntos, velocidad y radio de curvatura entre otros. La lista completa de estos datos, así como el método seguido para obtenerlos está explicado en el capítulo 3.1.



Figura 3.7: Módulo mathsGPS

Como datos de salida este módulo devolverá una nueva lista de *trackpoints*, que incluye todos los datos anteriores y los calculados por el propio módulo.

Módulo vectors

Se trata de un módulo que realmente tiene función de biblioteca, ya que contiene diversas funciones que operan con vectores y que son muy útiles para este proyecto. Algunas de esas funciones son:

- Pasar un vector en coordenadas polares a coordenadas cartesianas y viceversa

3 Diseño

- Rotar un vector un cierto número de grados
- Pasar grados a radianes y viceversa
- Sumar y restar vectores
- Obtener componentes del vector: Módulo, ángulo, etc...

Será un módulo principalmente utilizado por *mathsGPS* para realizar operaciones más complejas, tales como la obtención de la aceleración radial y tangencial entre otras.

Módulo classifier

El módulo *classifier* se encargará de dividir y clasificar la lista de puntos proporcionada como entrada en tramos o segmentos. Estos a su vez pueden ser curvas o rectas, dependiendo de distintos factores como el radio de curvatura o la variación en la dirección. Actualmente este módulo tendrá en cuenta este último factor, ya que como se verá en el capítulo 4, se confirma definitivamente que el radio de curvatura no es un dato fiable para clasificar tramos.



Figura 3.8: Módulo classifier

Este módulo devolverá una lista de listas. Cada lista corresponderá a un tramo, que estará definida como curva ("CURVE") o recta ("STRAIGHT"). Dentro de esa lista estarán almacenados todos los *trackpoints* que corresponden a ese tramo.

Módulo userLogMsg.erl

Este módulo tiene una finalidad más dirigida al desarrollador, ya que genera un *log* de depuración en donde muestra los datos que *Teller* ha procesado junto con otra información relevante. Esto es debido a que son muchos los datos que el dispositivo GPS proporciona, y sin embargo muchos de ellos no son demasiado útiles para el proyecto, generando datos innecesarios que pueden complicar la depuración. Algunos de estos datos son los campos *Valid* o *RCR*, ya comentados en el capítulo .

3.4 Arquitectura del sistema

Módulo `kmlGenerator.erl`

Como ya se vio en el capítulo anterior, para poder visualizar los datos hace falta generar dos ficheros: Por un lado un fichero *kml* que contiene toda la información referente al trayecto, y un fichero *html* para conectarse a *Google Maps*. Este módulo por lo tanto es el encargado de generar el primero de los ficheros mencionados, recibiendo primero como entrada la lista de *trackpoints* actualizada y clasificada.

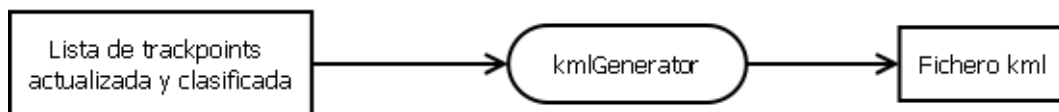


Figura 3.9: Módulo `kmlGenerator`

Módulo `htmlGenerator.erl`

Por último, el módulo *htmlGenerator* será el que genere el fichero *html* que se comunicará con *Google Maps* para que acceda a los datos del fichero *kml* y así poder visualizar los datos en el mapa.

Como ya se mencionó en el capítulo anterior, el fichero *kml* tiene que ser accesible por *Google Maps* para que se pueda realizar la visualización.

4

Ejecución de pruebas y resultados obtenidos

Una vez se ha diseñado el sistema que se quiere programar junto con los datos que se quieren obtener y visualizar, se puede proceder a realizar pruebas con datos reales y estudiar los resultados.

4.1 Ejecución de Teller

La invocación de Teller se realiza por terminal y es la siguiente

```
.\teller input_path output_name output_path url
```

- `input_path`: Ubicación del fichero *csv* que *Teller* va a procesar
- `output_name`: Nombre de los ficheros de salida (sin las extensiones *kml* o *html*)
- `output_path`: Ubicación de los ficheros de visualización generados por *Teller*

4 Ejecución de pruebas y resultados obtenidos

- url: Dirección pública donde estará el fichero *kml* al que *Google Maps* tiene que acceder

Si no se produce ningún error el resultado debería ser similar al que se muestra en la figura 4.1

```
*****
TELEMETRY PROJECT
v0.50a
*****

+INPUT FILE PATH: '../test/test001-in.csv'

+PARSING FILE...      DONE

+MAKING CALCULATIONS... DONE

+CLASSIFYING...      DONE

+GENERATING LOG FILES...
|- FILE 'maths_log'...  DONE
|- FILE 'tech_log'...  DONE
|- FILE 'user_log'...  DONE
\+ PATH FOR LOG FILES IS '../test' (FROM CURRENT DIRECTORY)

+GENERATING VISUALIZATION FILES...
|- KML FILE...         DONE
|- HTML FILE...        DONE

**SUCESS**
sergiog@babel:~/repos/teller/code/src$
```

Figura 4.1: Ejecución correcta de Teller

Al finalizar una ejecución satisfactoria *Teller* crea 5 ficheros. Tres de ellos son los logs de depuración *math_log*, *user_log* y *tech_log* con información sobre los datos procesados. Los dos restantes son los ficheros *kml* y *html* necesarios para la visualización en *Google Maps*, siendo este último el que hay que abrir.

4.2 Información sobre las pruebas realizadas

Para poder realizar las pruebas de ejecución se han utilizado logs de tres recorridos reales para que los procese *Teller*, y a partir de los resultados obtenidos comprobar la precisión del dispositivo GPS y del propio programa.

Estos tres recorridos estarán en formato *csv*, generados por el programa BT747 después de descargarlos de un dispositivo GPS modelo **i-Blue 747A+** como ya se explicó en el capítulo 3.4.2.

La información sobre cada recorrido se detalla a continuación

- Recorrido 1: Trayecto de unos pocos kilómetros por una autopista de Madrid sin cambios bruscos en la dirección ni en la velocidad
- Recorrido 2: Varias vueltas al circuito del Jarama, con curvas cerradas y cambios de velocidad
- Recorrido 3: Recorrido por una carretera secundaria a las afueras de Madrid con curvas cerradas y cambios de velocidad.

4.3 Estudio sobre los resultados obtenidos

En este apartado se realizará un estudio sobre los resultados obtenidos por *Teller* después de haber realizado las pruebas descritas en la sección anterior.

Por un lado se compararán los datos que proporcionen tanto el propio programa como el dispositivo GPS (datos como la distancia y la velocidad entre otros), para a partir de ahí determinar el grado de precisión de este último.

Hay que destacar que aunque se realicen tres pruebas, los resultados que se muestren serán aquellos en donde las discrepancias sean más llamativas y de esta forma se puedan sacar conclusiones respecto a los datos y a la precisión del dispositivo.

4.3.1 Distancia entre puntos

Como ya se detalló en el capítulo de diseño, el cálculo de la distancia entre dos puntos consecutivos que realiza *Teller* se basa en la fórmula **HaverSine**, que mediante la latitud y la longitud de los dos puntos devuelve la distancia entre ellos.

4 Ejecución de pruebas y resultados obtenidos

Distancia (metros)	
GPS	Teller
8.07	8.07862727022764
8.03	8.038063246605615
7.99	7.99652533204774
7.95	7.960651960618394
7.89	7.895612581356221
7.82	7.830323114250183
7.81	7.822979119848029
7.78	7.78446037431831
7.76	7.76460222406825
7.78	7.78883734145151

Cuadro 4.1: Tabla comparativa de distancias (Recorrido 2)

Por otro lado no se conoce el método que sigue el propio dispositivo GPS para realizar el cálculo, pero sí se pueden cotejar ambos datos. Esta comparación se muestra en el cuadro 4.1

Como puede comprobarse, los datos proporcionados por el dispositivo GPS y *Teller* son muy parecidos entre sí, lo que sugiere que el algoritmo interno del primero es similar al que usa *Teller*.

4.3.2 Velocidad

La tabla 4.2 muestra las velocidades obtenidas en el recorrido 3.

Esta vez se pueden observar discrepancias entre los datos proporcionados por ambos, suficientes para descartar errores de redondeo.

Después de haber realizado un estudio más profundo sobre este dato, se pueden llegar a varias conclusiones sobre la recogida de datos por el dispositivo GPS y posteriores cálculos.

Por un lado, el cálculo de la velocidad que realiza el dispositivo GPS no se basa en las posiciones geográficas obtenidas. Esto se puede afirmar porque *Teller* sí calcula la velocidad en base a las posiciones geográficas de dos puntos y a partir de la frecuencia de la toma de datos; y sin embargo como se puede comprobar en la tabla 4.2, los valores no coinciden. **Es lógico pensar por lo tanto que el dispositivo GPS tiene un método diferente de calcular la velocidad, que no está**

4.3 Estudio sobre los resultados obtenidos

Velocidad (km/h)	
GPS	Teller
90.054	90.4185481908532
93.048	94.24796342500449
96.2	95.6671600762328
99.19	98.04141931194849
100.687	99.6496065836903
103.937	100.80382810043352
105.078	101.4060158545924
109.729	106.49527631442942
110.027	107.3698442016365
112.006	107.94817596326352

Cuadro 4.2: Tabla comparativa de velocidades (Recorrido 3)

relacionado con la captura de datos geográficos, y que en el caso de que estos tuviesen un error de varios metros, la velocidad en principio no tendría que verse afectada por ello.

4.3.3 Dirección

Dirección (grados)	
GPS	Teller
354.457672	356.6172406522953
357.456268	1.0849953592847332
359.885406	2.9576913292985823
359.885406	6.7668546501982005
359.885406	11.365809841345364
359.885406	11.814374778646819
11.596088	19.42658387447174
15.422099	20.975028451395644
18.260235	22.842226711048674
20.082338	21.773616060612067

Cuadro 4.3: Tabla comparativa de direcciones (Recorrido 3)

Si bien los datos proporcionados por el dispositivo GPS y *Teller* en el cuadro

4 Ejecución de pruebas y resultados obtenidos

4.3 son parecidos entre sí, se puede detectar que hay un problema mucho más grave relacionado con la toma de datos. Si se observa los datos recogidos por el dispositivo GPS desde la tercera fila de la tabla hasta la sexta inclusive, se puede comprobar que **el valor de la dirección durante cuatro capturas no varía en absoluto**.

Obviamente se trata de un error, ya que aunque la motocicleta se moviera en una línea recta el valor seguiría variando. No solo porque el hecho de trazar una línea recta perfecta sería muy difícil de realizar por el piloto, sino por el propio margen de error del dispositivo GPS que probablemente produzca cambios de décimas o milésimas en esos valores. Estos dos factores hacen que sea prácticamente imposible que se mantenga exactamente la misma dirección entre dos tomas de datos consecutivas, y mucho menos si se trata de cuatro tomas consecutivas como ocurre en este caso.

Uno de los posibles motivos que puede haber detrás de este error tiene que ver con la capacidad del GPS para capturar los datos. Cada 200 ms una interrupción de reloj llega al dispositivo, que necesita cierto tiempo para realizar la captura; así como los cálculos internos y obtener todos los datos que luego almacena. Si por cualquier razón el proceso se demorara, se podría dar el caso de que el dispositivo no terminara de calcular todos los datos, dejando aquellos menos relevantes sin actualizar.

Obviamente esto se trata de una hipótesis, ya que no se conoce el funcionamiento interno del GPS, ni mucho menos los motivos exactos por los que ocurren estos incidentes. Pero en cualquier caso **lo que sí se puede afirmar es que los valores de la dirección proporcionados por el dispositivo pueden quedarse "congelados", estando desactualizados durante varias capturas de datos**.

4.3.4 Radio

Una de las formas que se habían propuesto para determinar la curvatura de los tramos era calculando el radio de la circunferencia circunscrita que pasaba por tres puntos consecutivos, como ya se explicó en el apartado 3.1.4. De esta forma, cuanto menor fuera el radio, más pronunciada sería la curva; y en el caso contrario, cuanto mayor fuera el radio, más se asemejaría a una recta.

Este método sin embargo ya levantaba sospechas sobre su efectividad desde el comienzo, ya que las curvas que toma una motocicleta en un trayecto real varían constantemente, y con ello los valores del radio en cada punto. Estas sospechas se confirmaron cuando se obtuvieron los resultados de *Teller* con los

4.4 Visualización de las pruebas

trayectos de prueba. En el cuadro 4.4 se muestran los valores del radio obtenidos en tramos de curva:

Radio de curvatura (metros)
1063.1080470219
2861.3566633898354
3258.4737709811848
228.4106508931037
1507.47967339963
624.4831015087267
448.90300626072013
1224.8064425026755
1712.8805356102548
979.4631090274997

Cuadro 4.4: Radios de curvatura de puntos consecutivos (Recorrido 2)

Como puede comprobarse, los valores del radio varían bruscamente de un punto a otro, cuando lo lógico sería que se tratara de cambios suaves. Por lo tanto se puede confirmar que **los valores del radio calculados por Teller no son fiables**, y que cualquier cálculo que se quiera realizar en base a ellos no serán correctos.

4.4 Visualización de las pruebas

Para finalizar este capítulo se muestra la visualización de los tres recorridos en *Google Maps*. Debido a que los marcadores que muestran la información de cada *trackpoint* están muy próximos entre sí (ya que el dispositivo GPS tiene una frecuencia de captura de 200 ms), el trazado que clasifica el recorrido no es muy visible. Es por esto que se ha decidido dividir esta última sección en dos grupos: Por un lado mostrando sólo el trazado del recorrido junto con los tramos clasificados, y por otro lado mostrando todos los marcadores con la información sobre cada *trackpoint*.

4 Ejecución de pruebas y resultados obtenidos



Figura 4.2: Trazado sin marcadores del recorrido 1

4.4 Visualización de las pruebas

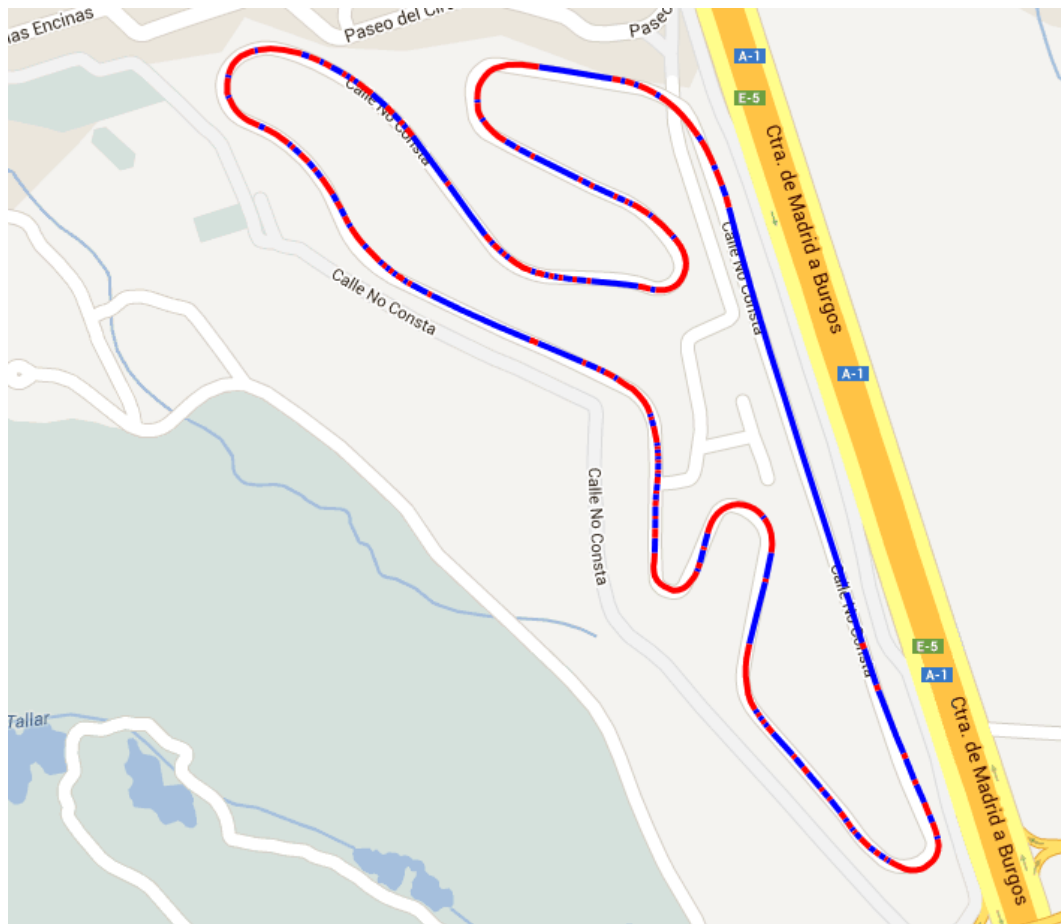


Figura 4.3: Trazado sin marcadores del recorrido 2

4 Ejecución de pruebas y resultados obtenidos

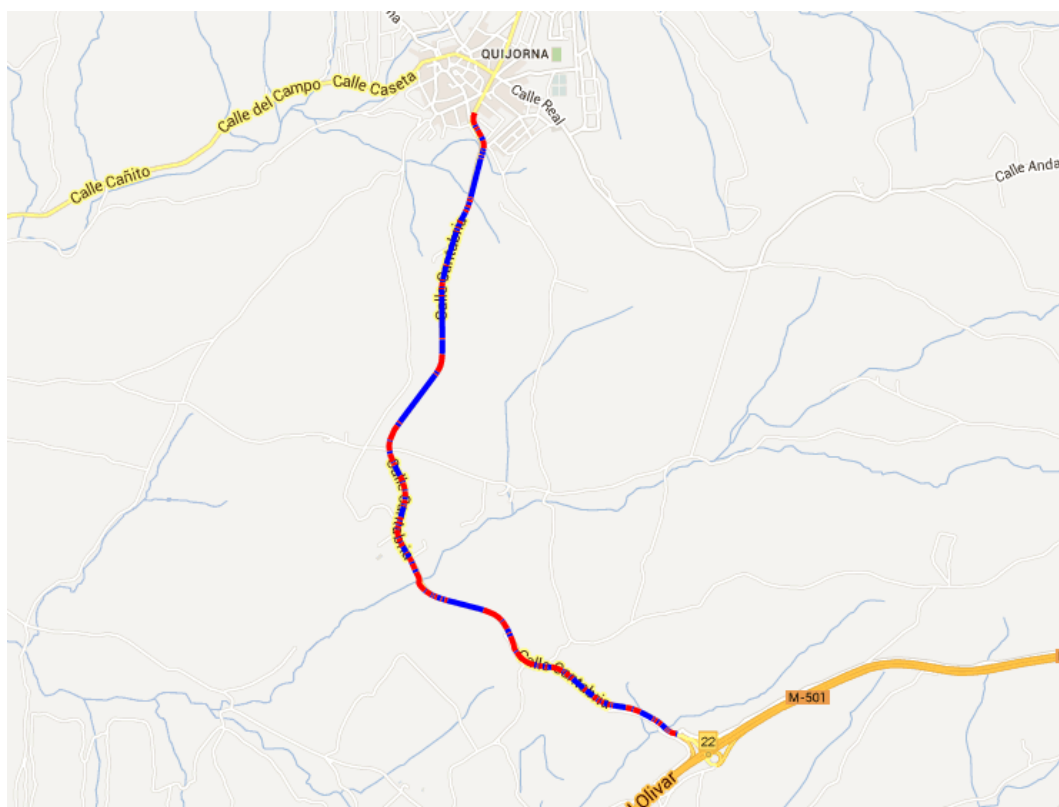


Figura 4.4: Trazado sin marcadores del recorrido 3

4.4 Visualización de las pruebas



Figura 4.5: Trazado con marcadores del recorrido 1

4 Ejecución de pruebas y resultados obtenidos



Figura 4.6: Trazado con marcadores del recorrido 2

4.4 Visualización de las pruebas

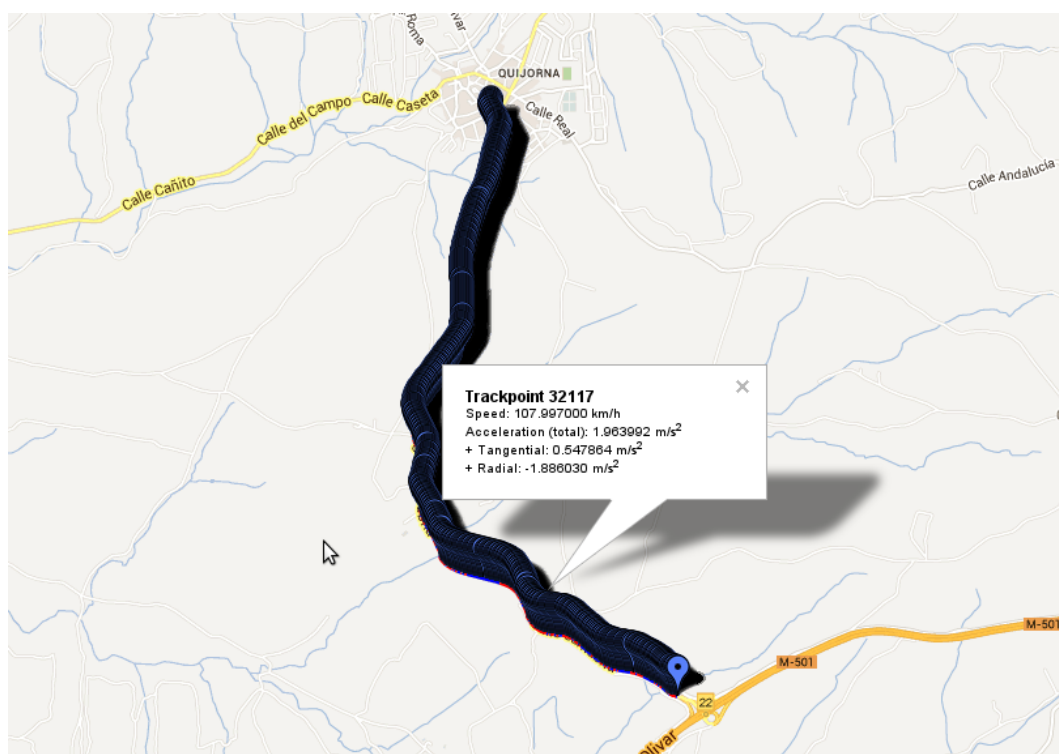


Figura 4.7: Trazado con marcadores del recorrido 3

5

Conclusiones

Como ya se ha podido comprobar, la funcionalidad adicional que se le puede dar a los dispositivos GPS usando la telemetría es muy amplia y a medida que la tecnología de estos vaya avanzando se podrán hacer muchas más cosas.

Mediante la creación de *Teller* se han podido obtener datos telemétricos que el propio dispositivo GPS no proporcionaba, datos interesantes que solo podían obtenerse con equipos profesionales de telemetría, generalmente usados en competiciones y no accesibles para aquellos pilotos novatos o amateurs que utilizan la motocicleta como hobby, y que tienen curiosidad por saber más sobre este campo, ya sea para conocer información telemétrica adicional sobre sus propios recorridos, o simplemente por diversión.

Por otro lado los datos que el dispositivo GPS almacenaba se ha podido comprobar cuales eran fiables y cuales no. Además se ha podido realizar una clasificación del recorrido en tramos, dependiendo de si se trataba de una curva o una recta. Y finalmente todos esos datos se han podido visualizar para que el usuario pueda ver datos sobre su recorrido: Velocidad que tenía en distintos puntos del trayecto, en qué parte de la curva aceleraba o frenaba y con qué intensidad, etc.

Sin duda las posibilidades que tiene la aplicación de la telemetría en el motociclismo son muchas, y es posible que con la creación de *Teller* se hayan sentado las bases para futuros proyectos que indaguen en el mismo campo.

5 Conclusiones

5.1 Trabajo futuro

Si bien Teller en esta versión inicial es capaz de calcular información telemétrica, clasificar tramos y visualizarlos; las posibilidades que aún puede ofrecer son muchas, y con el tiempo necesario se pueden ir implementando paulatinamente. Algunas de estas nuevas funcionalidades que podría ofrecer son:

- Análisis y mejora en la precisión de los datos: Mediante la aplicación de filtros que minimicen el ruido en este tipo de dispositivos, o el uso de técnicas de *smoothing* para una clasificación de recorridos más precisa.
- División automática de tramos cuando haya grandes lapsos de tiempo entre una captura y la siguiente o estos no sean significativos: Por ejemplo cuando se esté atravesando un túnel muy largo, se haya detenido la moto-cicleta o la velocidad sea tan baja que no sea adecuado que *Teller* realice los cálculos
- Nueva clasificación de tramos dependiendo del tipo de curva: Curvas abiertas, cerradas, chicanes; o teniendo en cuenta otros datos como la velocidad a la que se está tomando, etc.
- Servicio web para que los usuarios puedan almacenar sus recorridos y compararlos con otros usuarios

Los ejemplos mencionados son solo los más importantes por donde se puede avanzar a partir de los resultados obtenidos con *Teller*, pero no son los únicos. La realización de estos objetivos ayudará sin duda a progresar en el campo de la telemetría y muy posiblemente a que aparezcan muchas más funcionalidades aparte de las listadas, convirtiendo a *Teller* en el precursor de lo que podría ser una aplicación mucho más compleja y siempre orientada al motociclismo.

A

Dispositivo i-Blue 747A+

Como ya se mencionó brevemente en el capítulo 1.3, el modelo de dispositivo GPS utilizado en el proyecto es el **i-Blue 747A+** de la empresa *TranSystem Inc.*

Se muestra a continuación una descripción del producto extraída de la página web del fabricante (<http://www.transystem.com.tw/product.php?b=G&m=pe&cid=4&sid=&id=59>), que incluye los siguientes detalles técnicos (en inglés):

This BT GPS logger features an all-in-one, cost-effective portable GPS logging solution. It allows you to log your route by setting the interval of time/ distance/ speed. Point of interest can also be recorded by push of a button. Through user friendly utility, it shows



Figura A.1: Dispositivo i-Blue 747A+

A Dispositivo i-Blue 747A+

your track on Google Earth.

Key Features:

- Dual mode for both Data record and Navigation
- Push button to record data manually
- User can record the date by setting the interval of time, distance and speed
- 125000 waypoints
- Semi indoor
- AGPS available
- Fuzzy Auto on/off
- Photo Tagger software

Applications:

- Record your travels
 - Manage business trip expenses
 - Concerned about one's driving behavior
 - Record Point of Interest by a push button
 - Geo-photo

Category	Parameter	Description
General		
GPS Chip	MTK	
Frequency	L1,1575.42 MHz	
C/A Code	1.023MHz chip rate	
Channels	66 CH for tracking	
Antenna	Built-in patch antenna with LNA	
Datum	WGS-84	
Acquisition		
Cold Start	35 sec, average	
Warm Start	34 sec, average	
Hot Start	1.5 sec, average	
AGPS	<15sec	
Power		
Battery	Li-Ion	
Charging time	3hrs (Typical)	
Operation Time	32hrs	
Power Charge	Mini USB	
Environmental		
Operating	-10°C to +60°C	
Storage	-20°C to +60°C	

Charging 0°C to +40°C
 Relative Humidity 5% to 90% non-condensing
 Accuracy (none DGPS)
 Position Without aid: 3.0m 2D-RMS
 DGPS: 2.5m, 2D-RMS
 Velocity Without aid: 0.1m/s
 DGPS: 0.05m/s
 Sensitivity -165dBm (tracking)
 Dynamic
 Altitude <18,000m
 Velocity <515m/sec
 Acceleration 4g
 Interface
 Bluetooth version 1.2 compliant
 Output Power Class2 (10 meter in free space)
 Frequency 2.4 to 2.4835GHz
 Protocol
 NMEA-0183 (V3.01) - GGA, GSA, GSV, RMC (default)
 Data bit: 8, stop bit:1 (Default)
 Physical
 Dimension 46.5W x 72.2L x 20H mm
 Weight 64g
 Data Log
 Up to 125,000 way points.
 Log GPS data by time interval / distance / speed limit.
 Log GPS data by push button.
 Others
 LED Bluetooth, Navigation, Battery/ charger status
 Certification CE / FCC
 *Bluetooth is a trademark owned by Bluetooth SIG, Inc.
 *Specifications subject to change without prior notice.
 *Copyright© 2010 Transystem Inc. All Rights Reserved.
 *Citation MTK original chipset spec

B

Listado de dispositivos GPS compatibles con BT747

- i-Blue 747 (<http://www.transystem.com.tw>)
- i-Blue 747A+ (<http://www.transystem.com.tw>)
- i-Blue 757 (<http://www.transystem.com.tw>)
- i-Blue 821 (<http://www.transystem.com.tw>)
- Photomate 887 (<http://www.transystem.com.tw>)
- Photomate 887 Lite (<http://www.transystem.com.tw>)
- Qstarz BT-Q1000 (<http://www.qstarz.com>)
- Qstarz BT-Q1000X (<http://www.qstarz.com>)
- Qstarz BT-Q1200 (<http://www.qstarz.com>)
- Qstarz BT-Q1000eX GPS Lap Timer

B Listado de dispositivos GPS compatibles con BT747

- iTrek z1 (<http://www.itrekgps.com>)
- Konet BGL-32 (<http://www.konet-sys.com>)
- Gosget MB-688L
- Gosget MB-988SL
- Holux M-241 (www.holux.com) - the baud rate must be set at 38400 !!
- Holux GR-241 (www.holux.com) - baud rate must be 38400
- Holux GPSport 245 (GR-245) - (www.holux.com) The baudrate must be set to 38400 and the device type to 'Holux 245'
- Holux M-1000C ([holux.com](http://www.holux.com)) FCC RJIM-1000XX. The baudrate must be set to 38400 and the device type to 'Holux 245' // The protocol for this device is slightly different, so you can also adjust the protocol in the Settings menu
- Holux M-1200e (www.holux.com) - Log format type is 'Holux GR-245'
- Bluenext BN-900GR (www.bluenext.co.uk)
- Qstarz BT-Q818extreme (www.qstarz.com)
- Qstarz BT-Q1300 (Nano Q1300) - BT-Q1300s
- EBON DL-3200BT (www.ebontek.com)
- EBON DL-5100 BT (www.ebontek.com)
- Visiontac VGPS-900 (www.visiontac.com) - set the CSV file from the mini SDCard as input
- Blumax GPS-4043 (www.blu-max.eu or <http://www.eurotronic.net/>) / Equivalent to iBlue 747
- Blumax GPS-4044 (www.blu-max.eu or <http://www.eurotronic.net/>) / Equivalent to iBlue 747 A+
- Sanav ML-7 (sanav.com, report)
- BL-99 (seen on www.etek.com.tw - looks like iBlue 747)
- NCS-NAVI R120+ (www.ncsnavi.com) [website not very available]
- NCS-NAVI R150+ (www.ncsnavi.com) [website not very available]
- Pentagon Pathfinder P3106 (pentagram.pl)

- Royaltek MBT-1100 (www.royaltek.com) [see forum for some details regarding BT747 support]
- GiSTEQ Phototrackr CD111 (www.gisteq.com) [see forum for user feedback]
- GiSTEQ DPL700 PhotoTrackr Lite (www.wondeproud.com , www.gisteq.com)

Bibliografía

- [1] Dan Anderson. Defining waypoints, tracks, and routes. <http://www.gpsmap.net/DefiningPoints.html>.
- [2] Joe Armstrong. *Programming Erlang: Software for a Concurrent World*. Pragmatic Bookshelf, 2007.
- [3] Club Trail de Andalucía. Curso de navegación y GPS. Terminología y conceptos teóricos. <http://cursonavegacion-aweileitor.blogspot.com.es/2007/02/capitulo-i-terminologa-y-conceptos.html>.
- [4] Ángel Franco García. Relación entre las magnitudes angulares y lineales. <http://www.sc.ehu.es/sbweb/fisica/cinematica/circular1/circular1.htm>.
- [5] Ángel Herranz. Contramanillar. <http://contramanillar.wordpress.com/2012/03/03/contramanillar/>.
- [6] Ángel Herranz. Curvas, velocidad e inclinación. <http://contramanillar.wordpress.com/2012/03/03/contramanillar/>.
- [7] Ignacio López. Conceptos y terminología GPS. <http://www.tic2.org/Ignacio/GPS/Terminologia/Terminologia.htm>.
- [8] C. R. Nave. Basic vector operations. <http://hyperphysics.phy-astr.gsu.edu/hbase/vect.html>.
- [9] M. Olmo and R. Nave. Fuerza centrípeta. <http://hyperphysics.phy-astr.gsu.edu/hbasees/cf.html>.
- [10] Zemansky Sears. *Física universitaria*. Editorial Pearson 12ª edición, 2010.
- [11] Paul A. Tipler and Gene Mosca. *Física para la ciencia y la tecnología*. Editorial Reverté 5ª edición, 2010.

Este documento esta firmado por



Firmante	CN=tfgm.fi.upm.es, OU=CCFI, O=Facultad de Informatica - UPM, C=ES
Fecha/Hora	Fri Feb 14 19:54:00 CET 2014
Emisor del Certificado	EMAILADDRESS=camanager@fi.upm.es, CN=CA Facultad de Informatica, O=Facultad de Informatica - UPM, C=ES
Numero de Serie	630
Metodo	urn:adobe.com:Adobe.PPKLite:adbe.pkcs7.sha1 (Adobe Signature)